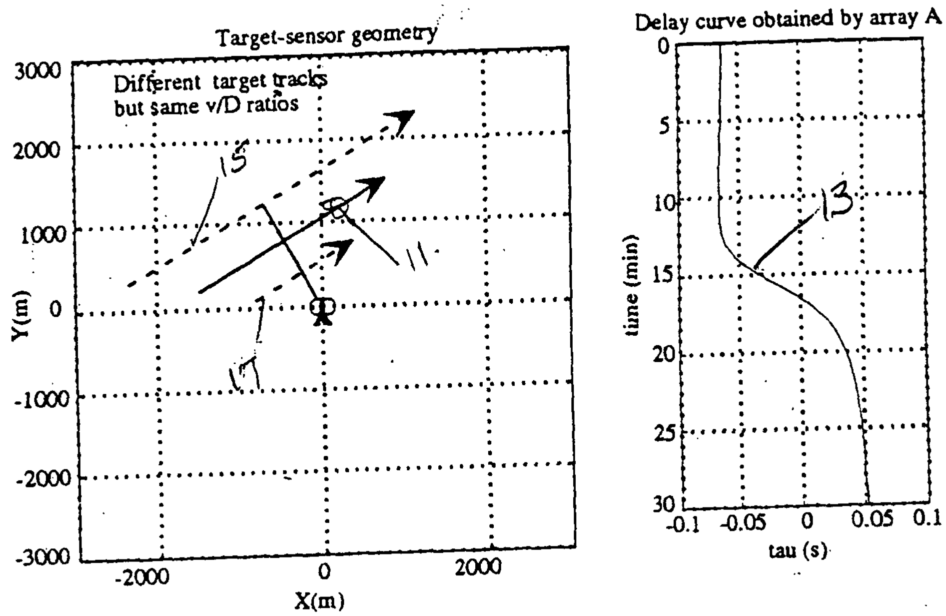


Figure 1a

Figure 1b

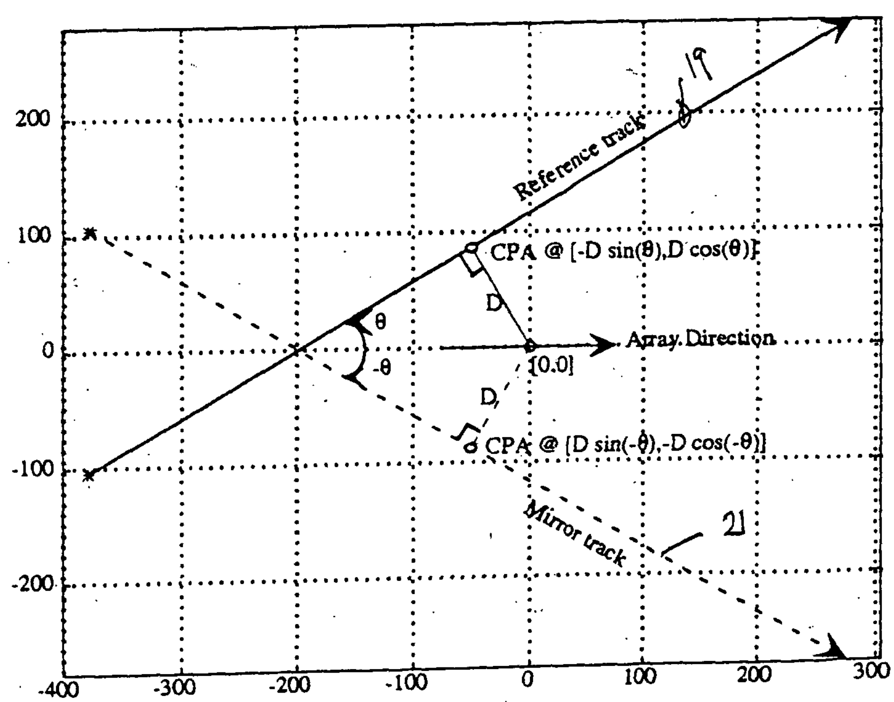


Figure 2

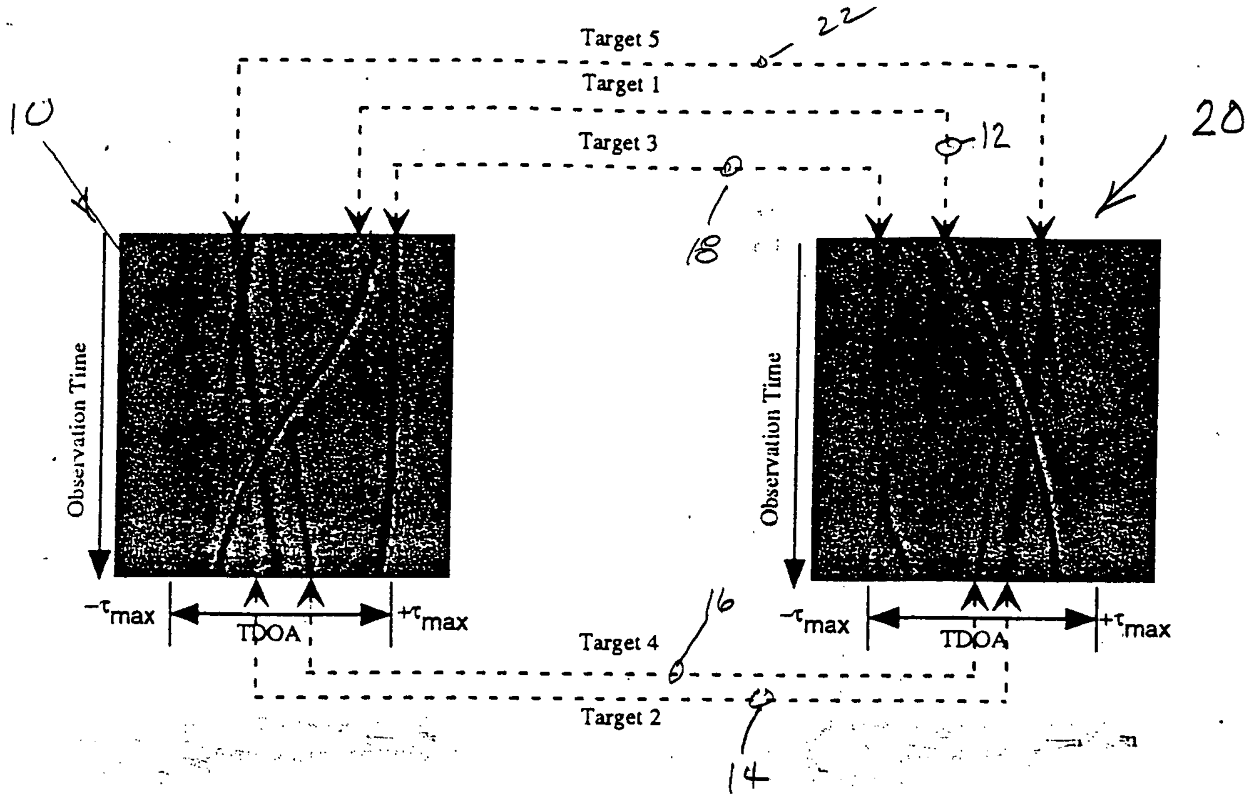
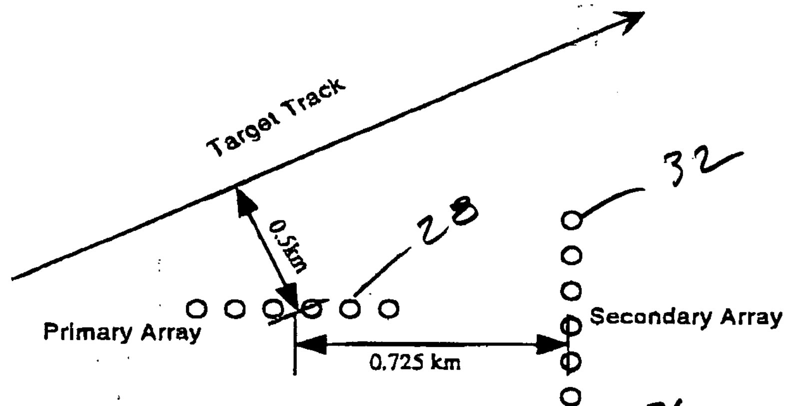


Figure 3 a

Figure 3 b

Figure 4a



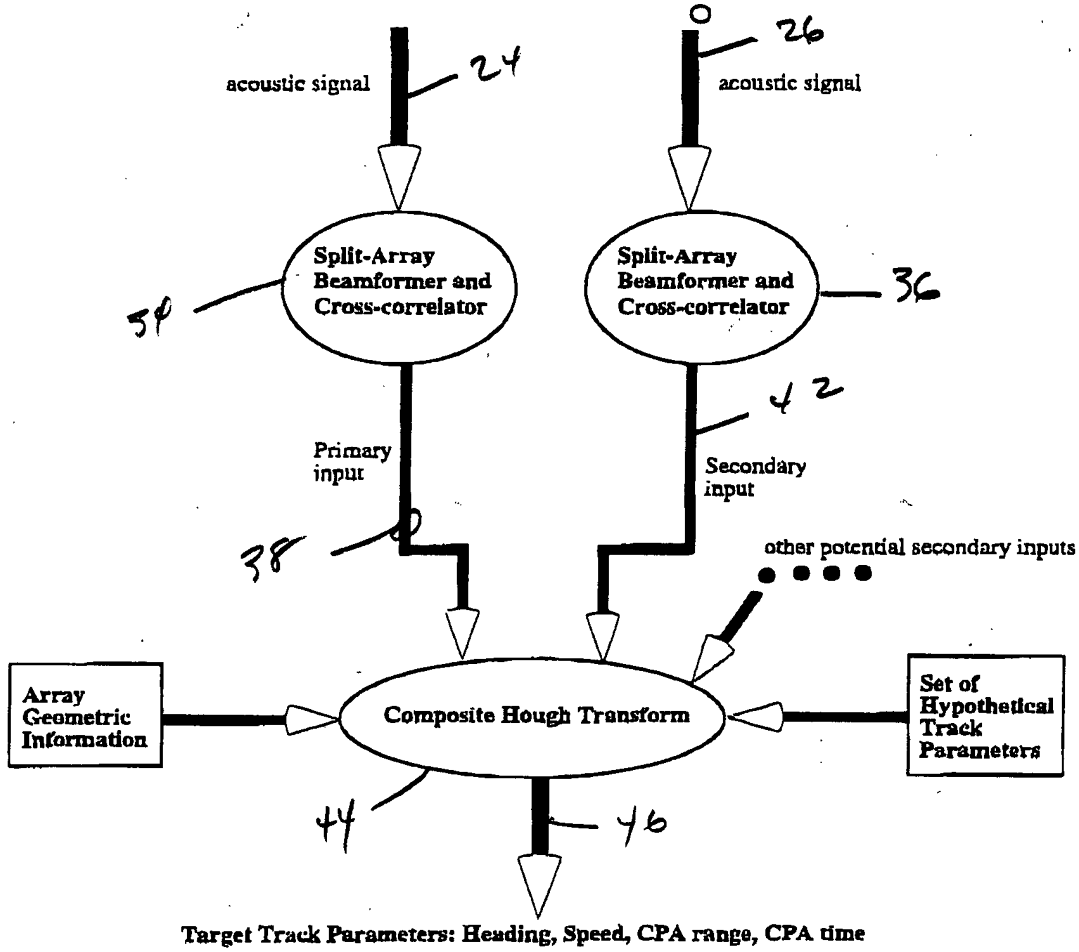


Figure 4b

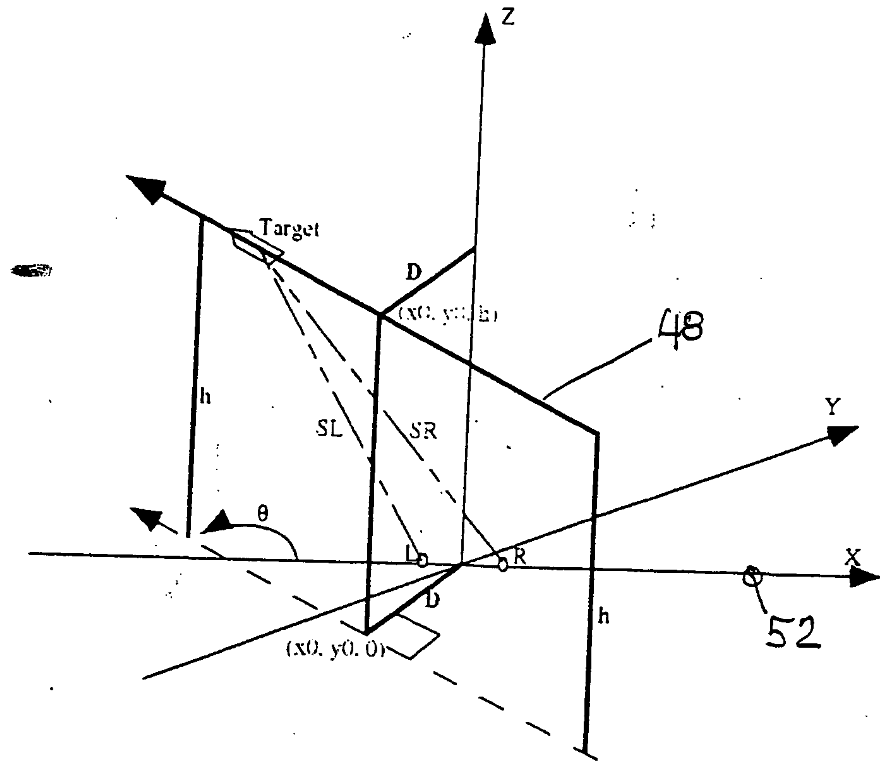
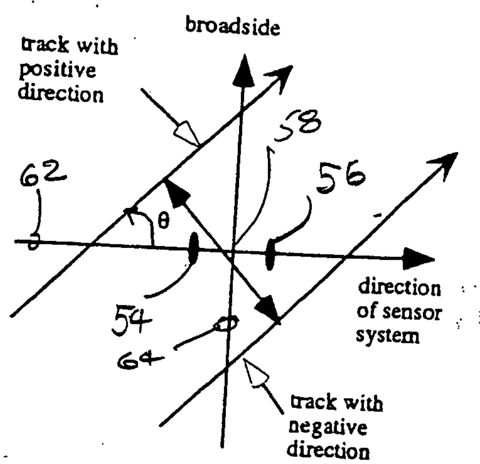
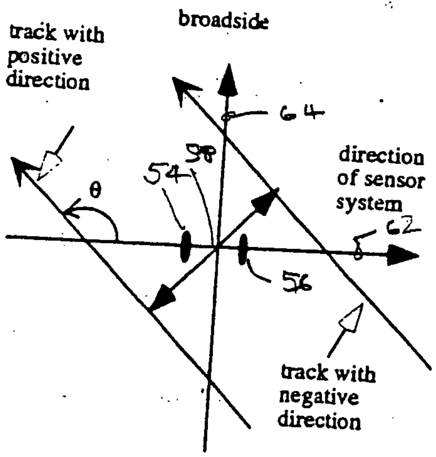


Figure 5



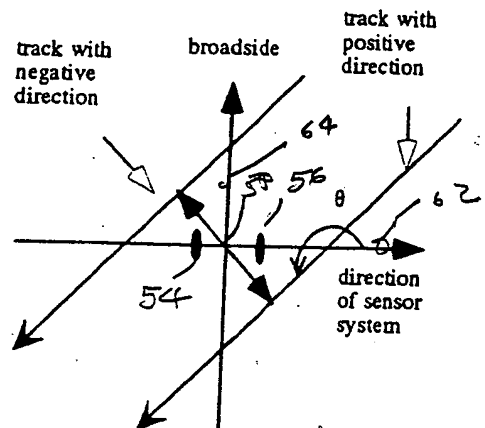
$0° < \theta < 90°$

Figure 6a



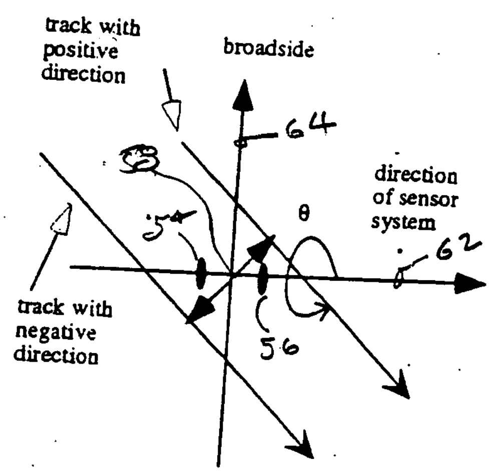
$90° < \theta < 180°$

Figure 6b



$180° < \theta < 270°$

Figure 6c



$270° < \theta < 360°$

Figure 6d

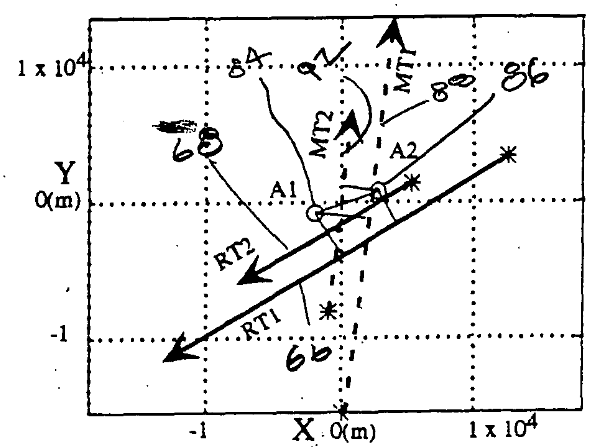


Figure 7a

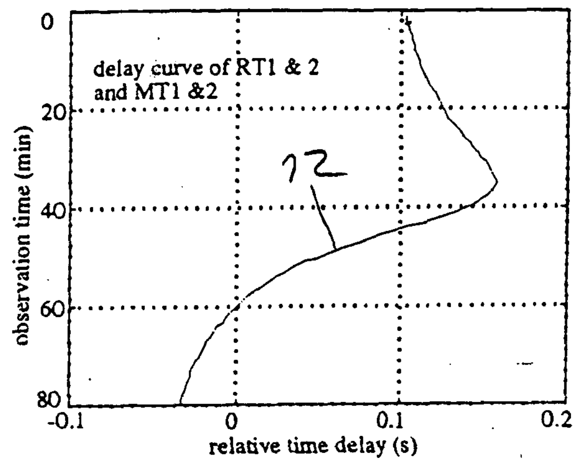


Figure 7b

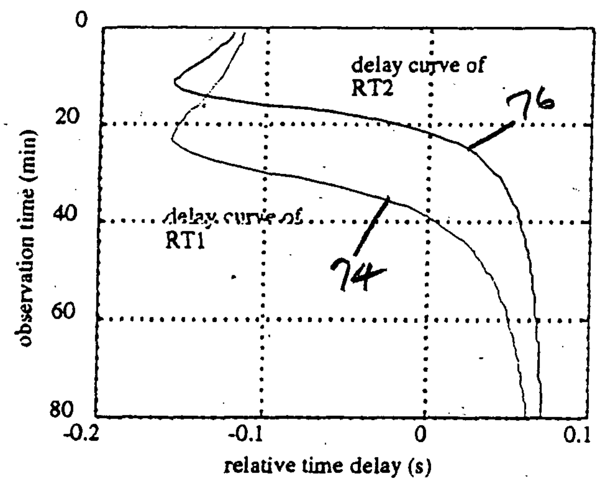


Figure 7c

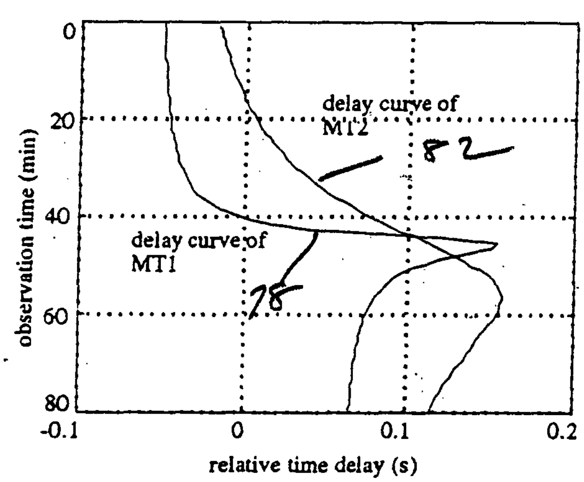


Figure 7d

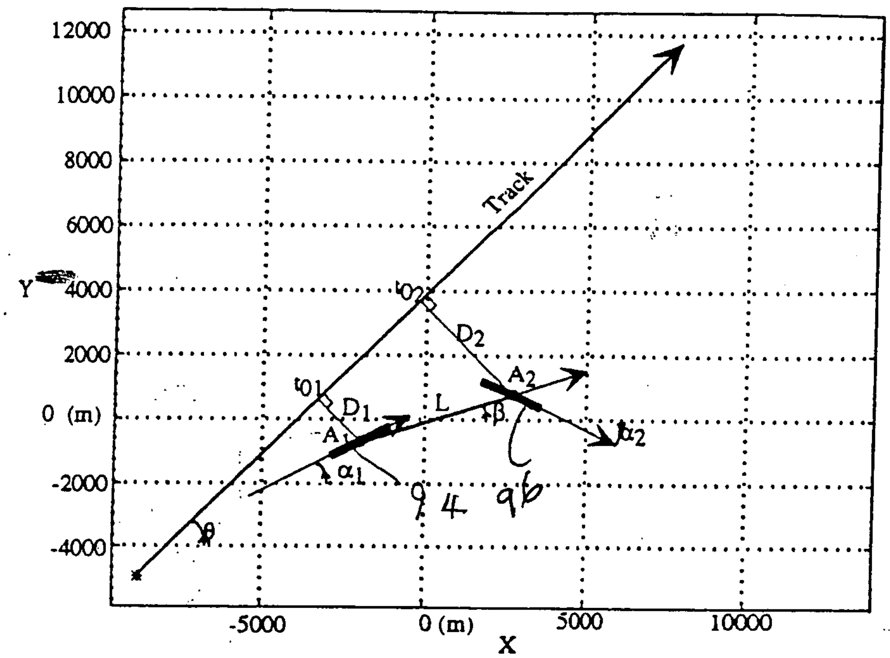


Figure 8

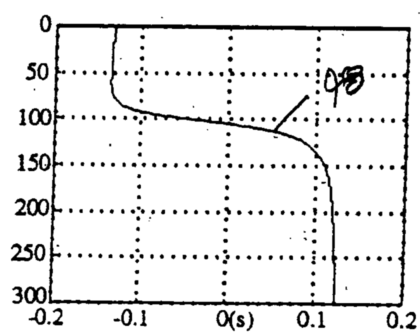


Figure 9a

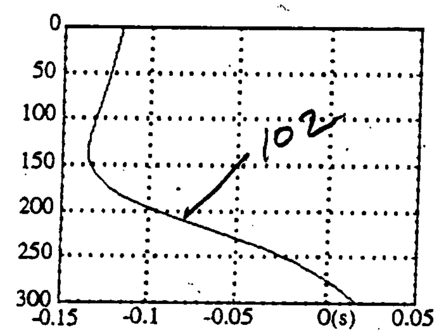


Figure 9b

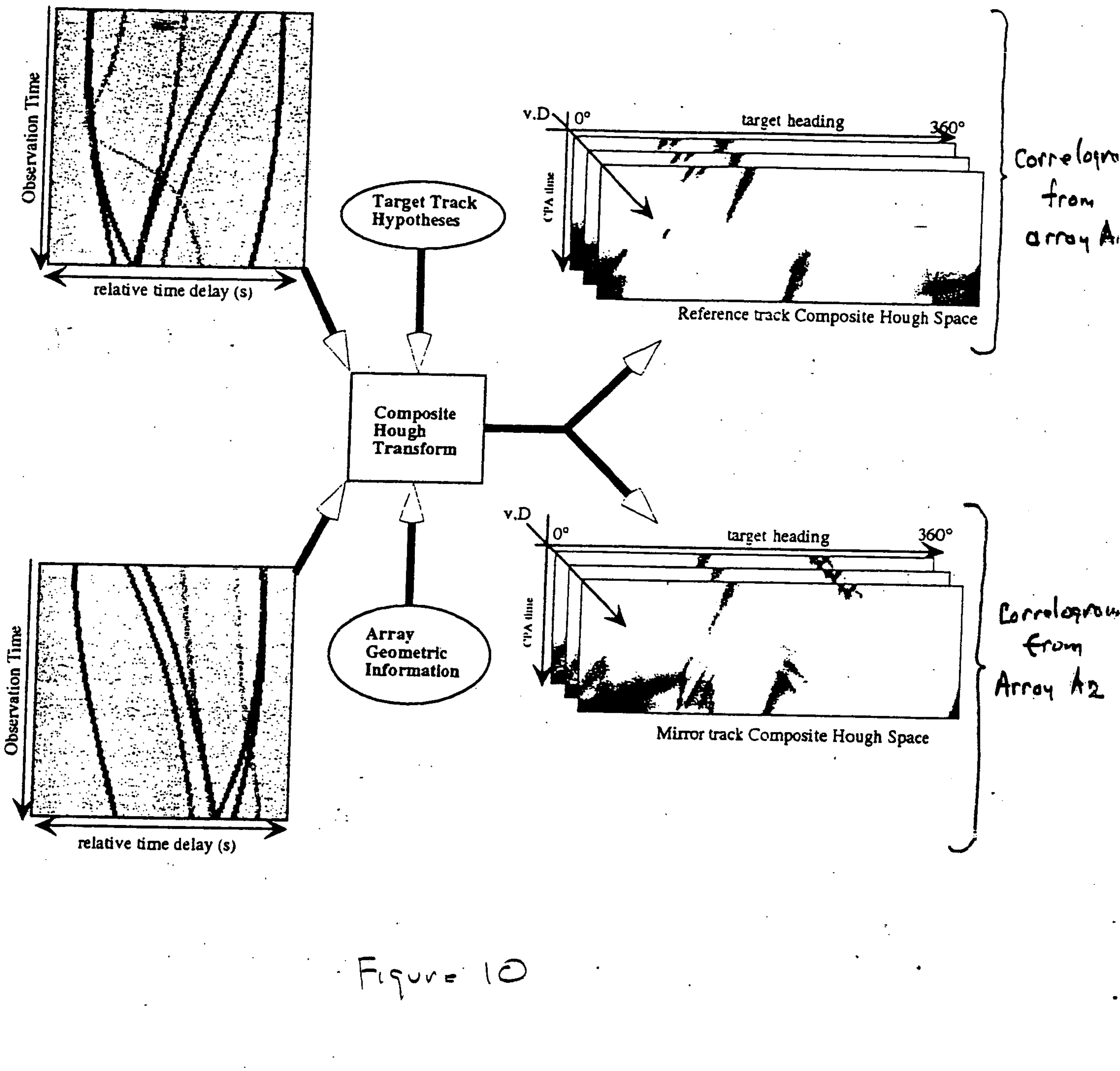

Observation Time
relative time delay (s)
Target Track Hypotheses
Composite Hough Transform
Array Geometric Information
Observation Time
relative time delay (s)
v.D
0°
target heading
360°
CPA time
Reference track Composite Hough Space
Correlogram from array A1
v.D
0°
target heading
360°
CPA time
Mirror track Composite Hough Space
Correlogram from Array A2

Figure 10

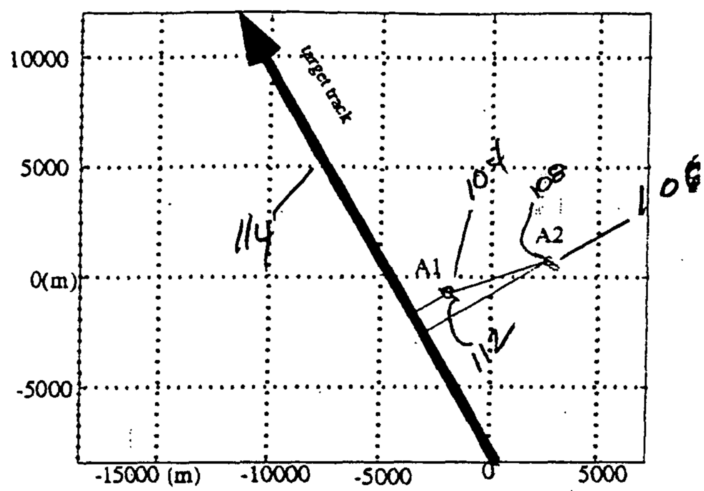


Figure 11a

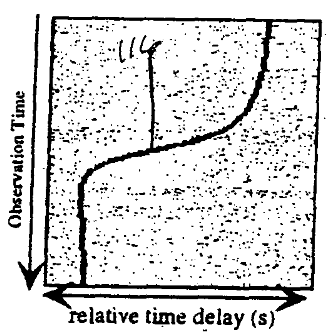


Figure 11b

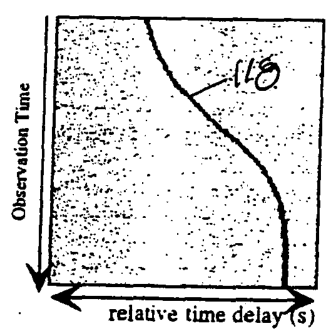


Figure 11c

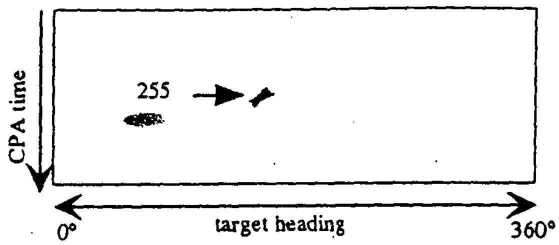


Figure 12a

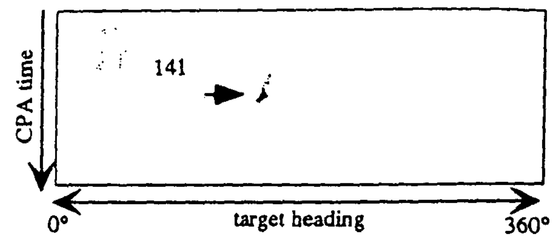
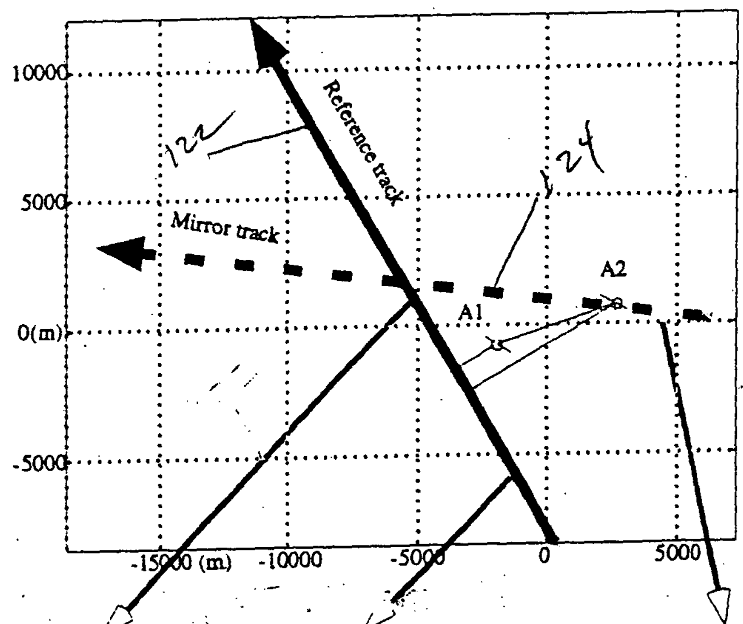


Figure 12b

Figure 13a



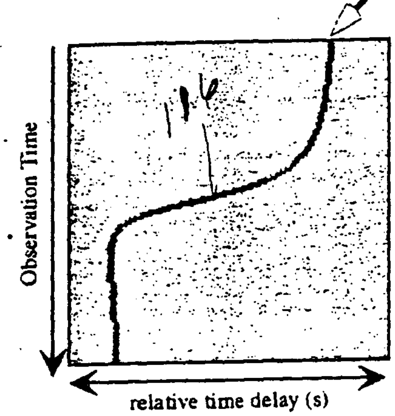


Figure 13b

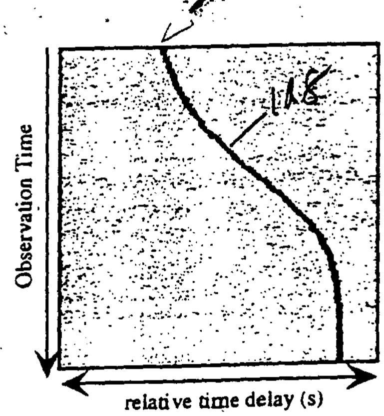


Figure 13c

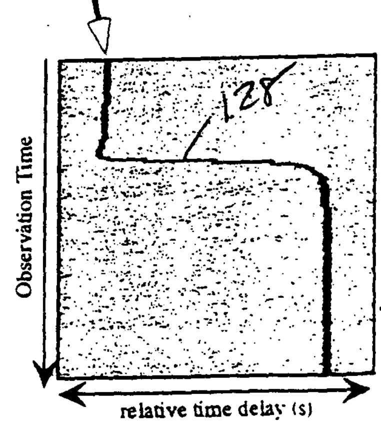
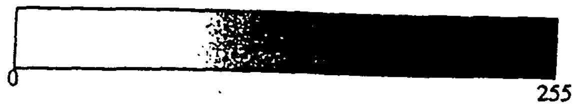


Figure 13d



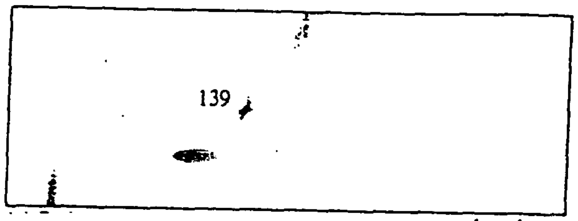


Figure 14a

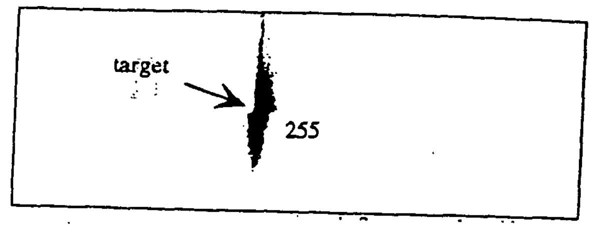


Figure 14b

Figure 15a

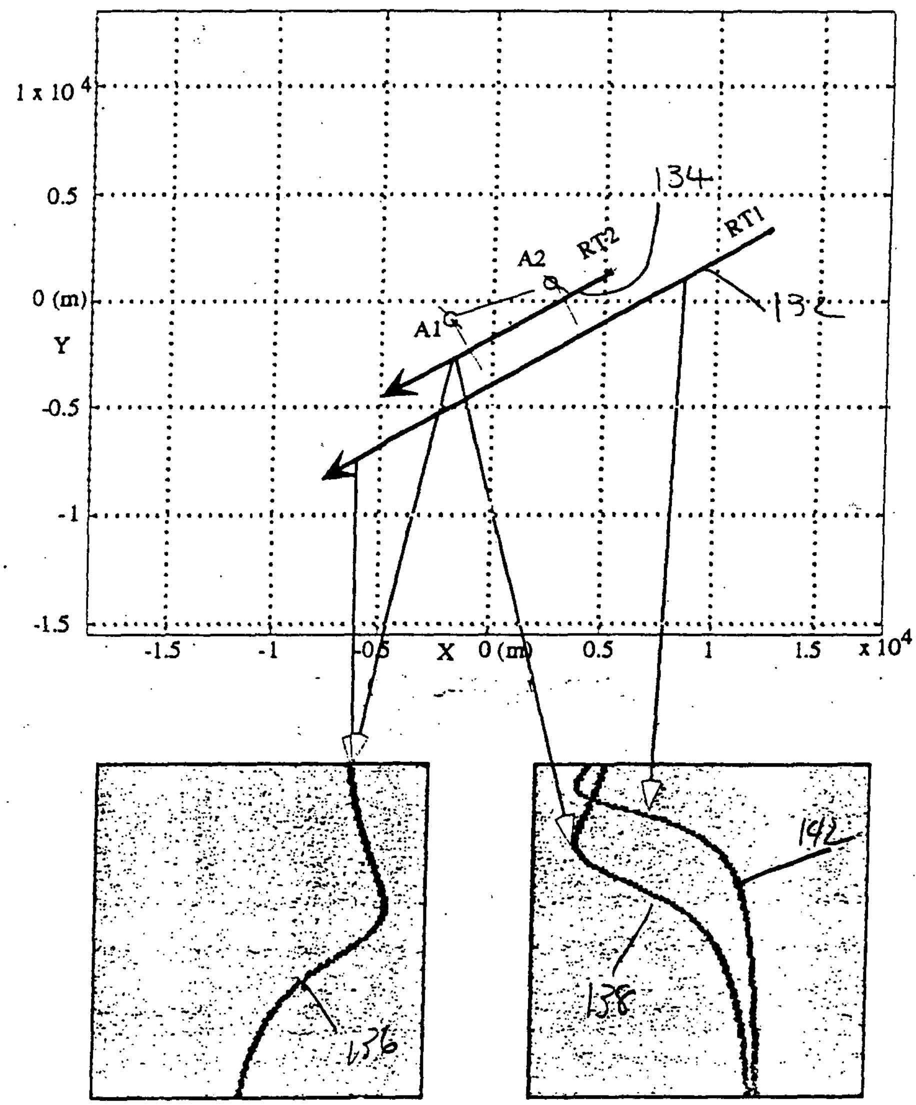


Figure 15b

Figure 15c

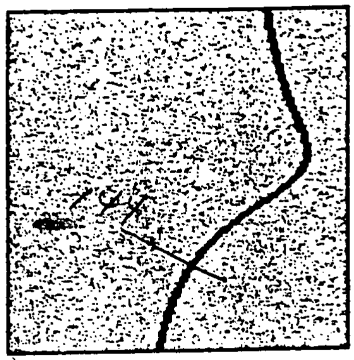


Figure 16a

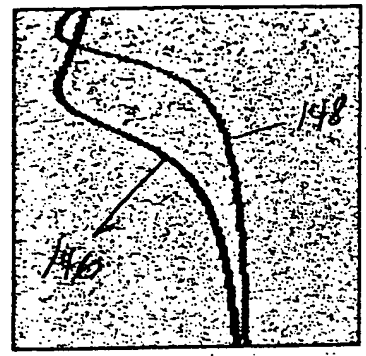


Figure 16b

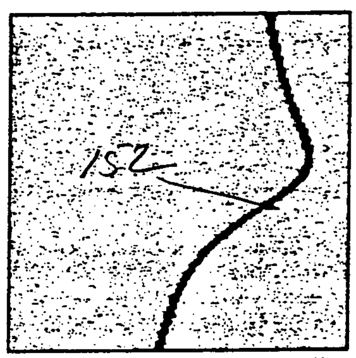


Figure 16c

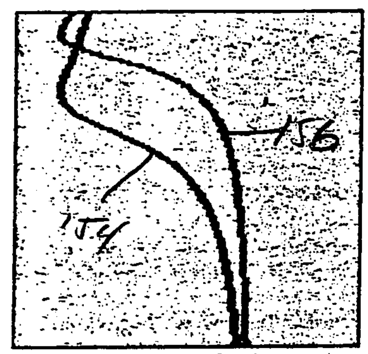


Figure 16d

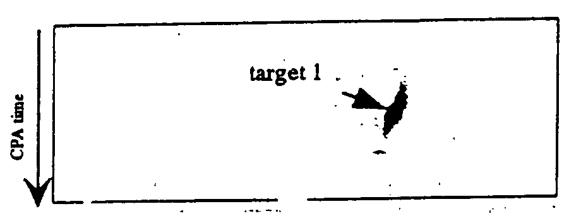


Figure 17a

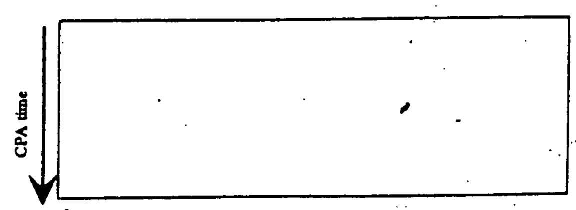


Figure 17b

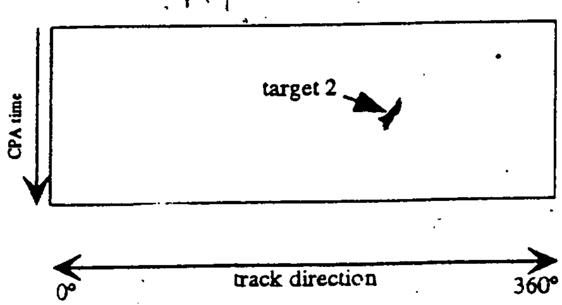


Figure 17c

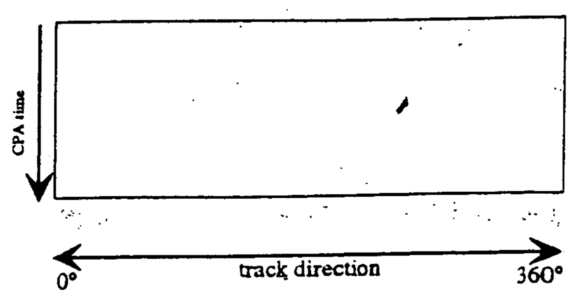


Figure 17d

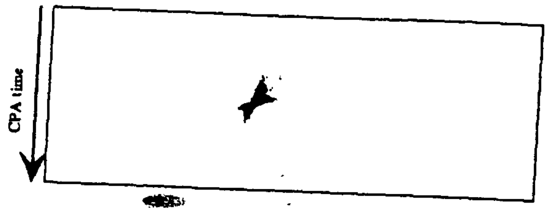


Figure 18a

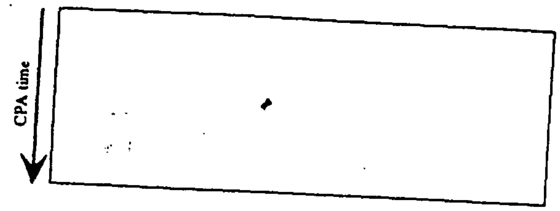


Figure 18b

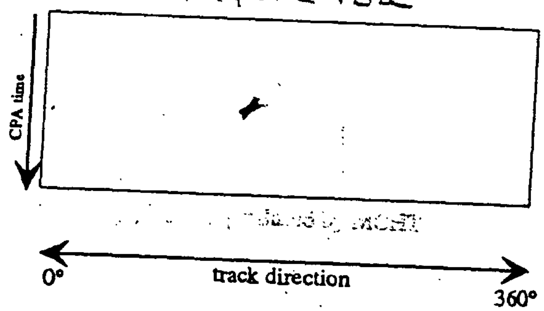


Figure 18c

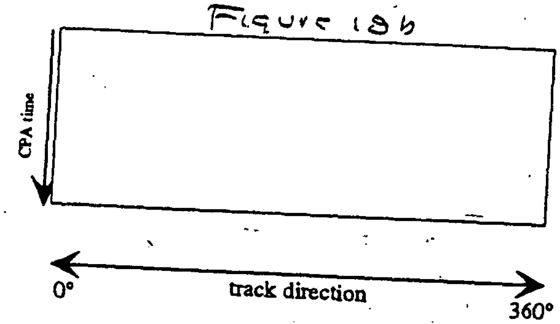


Figure 18d

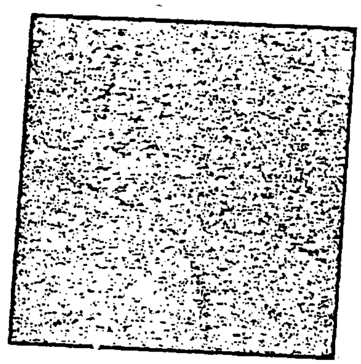


Figure 19a

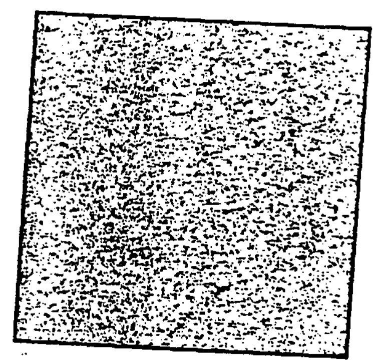


Figure 19b

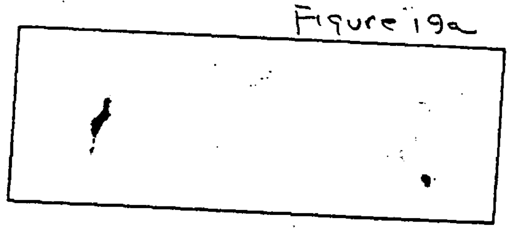


Figure 19c

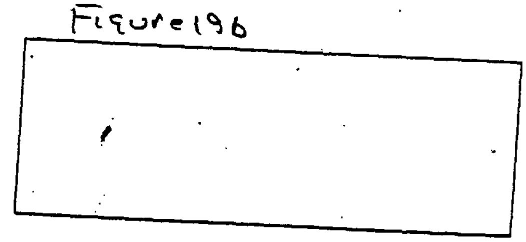


Figure 19d

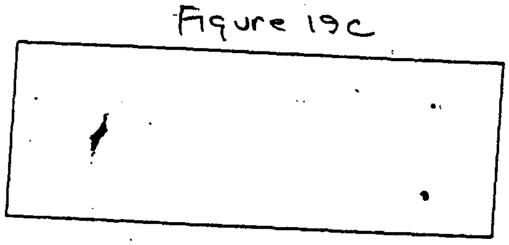


Figure 19e

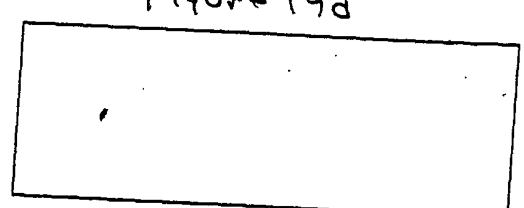


Figure 19f

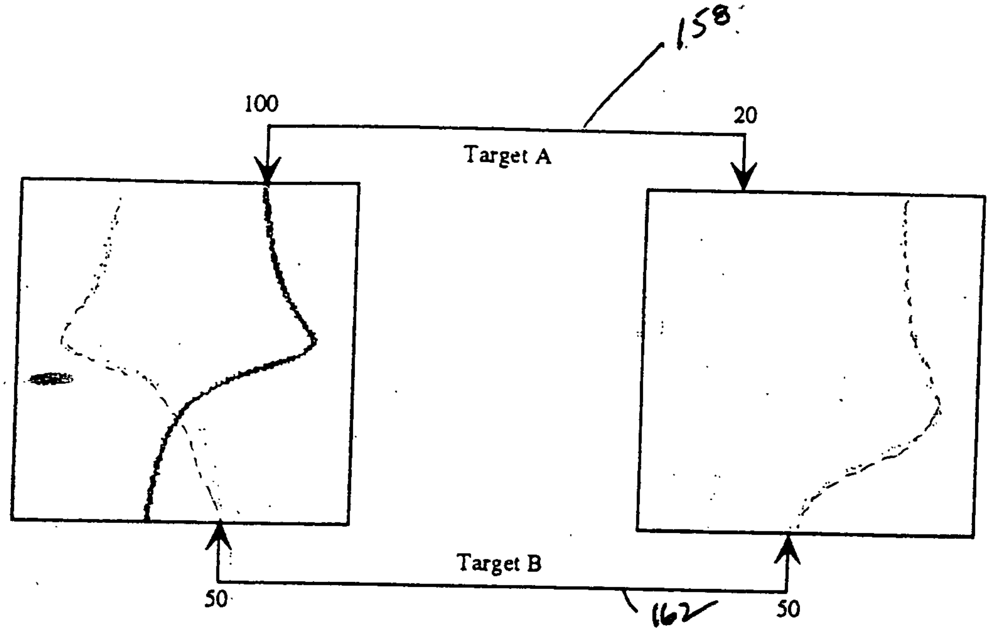


Figure 20

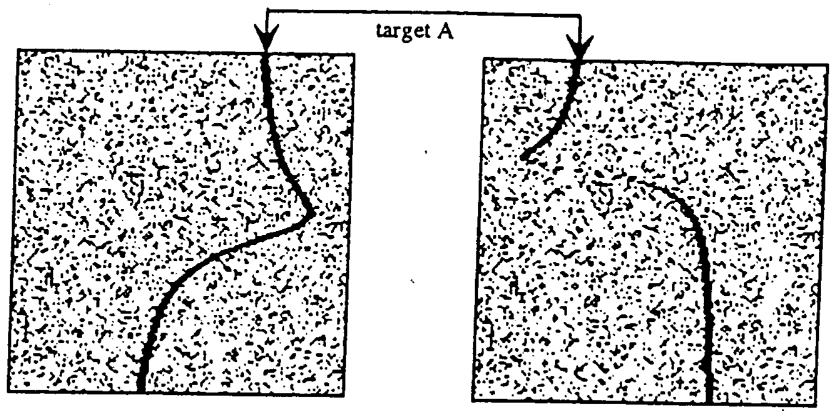


Figure 21a

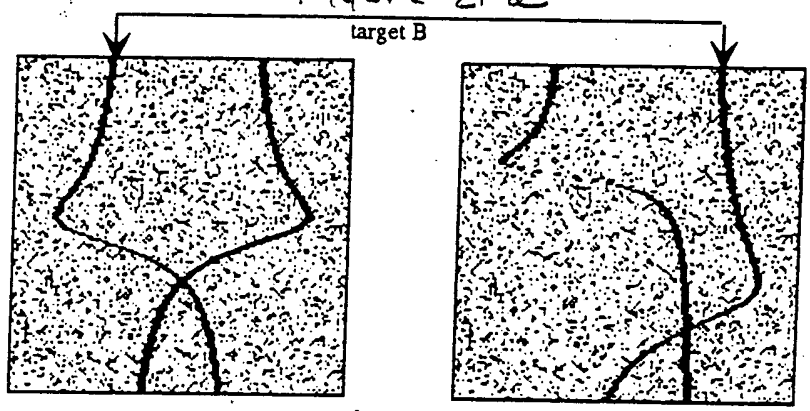


Figure 21b

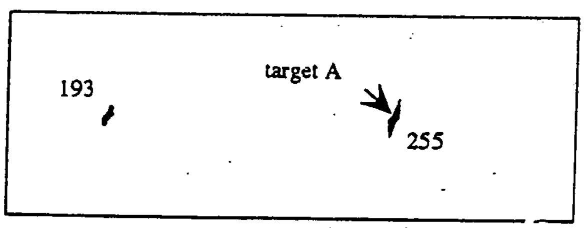


Figure 21c

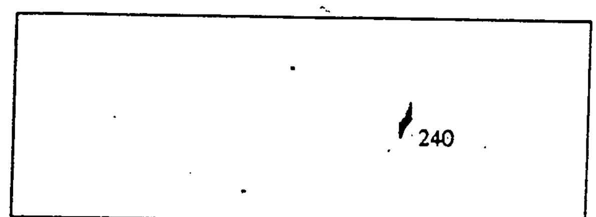


Figure 21d

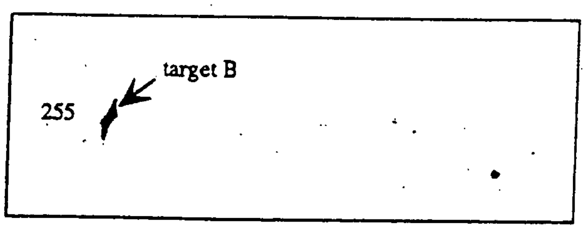


Figure 21e

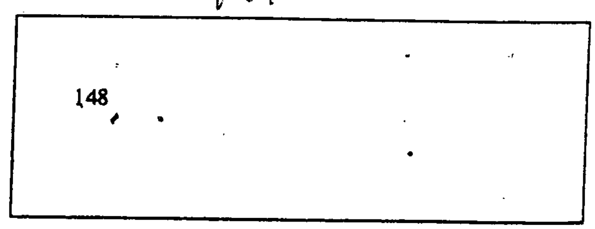


Figure 21f

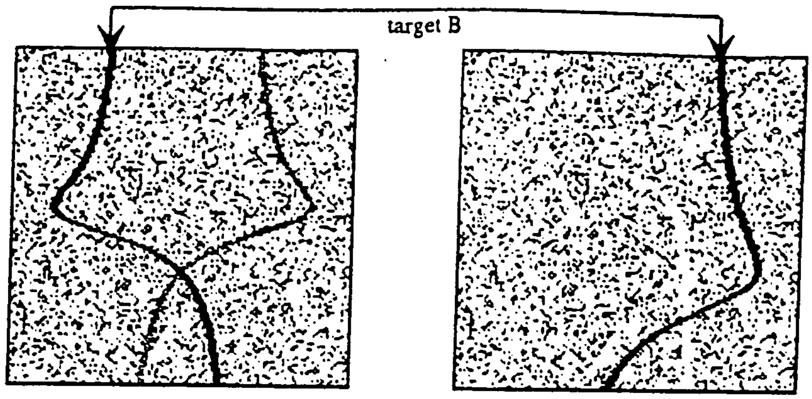


Figure 22a

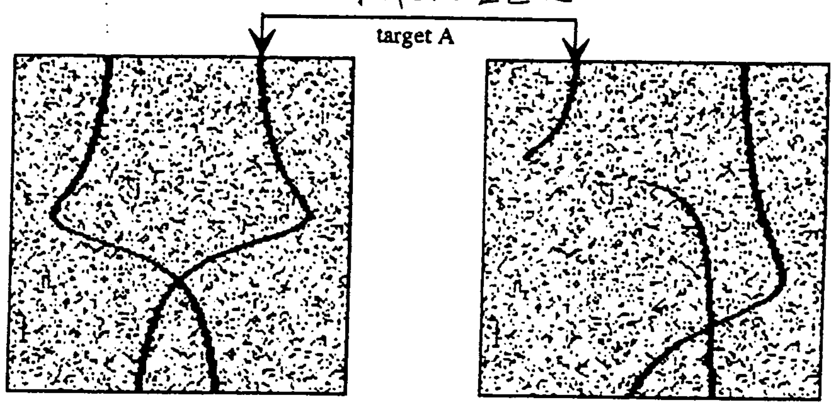


Figure 22b

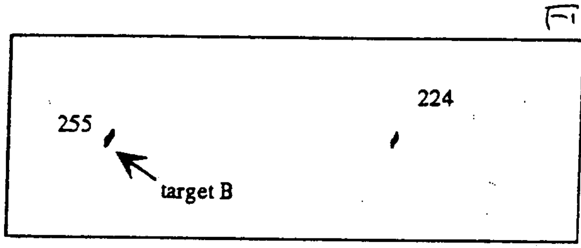


Figure 22c

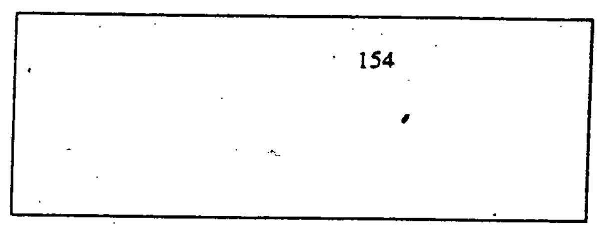


Figure 22d

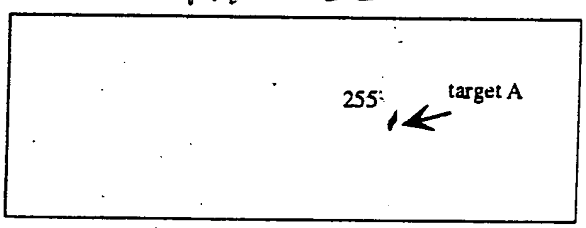


Figure 22e

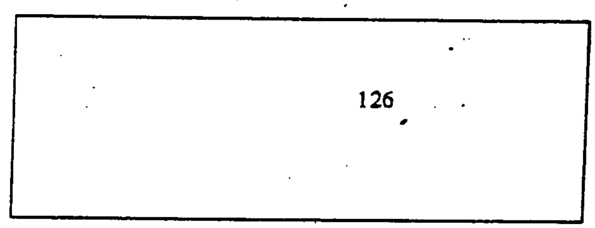


Figure 22f

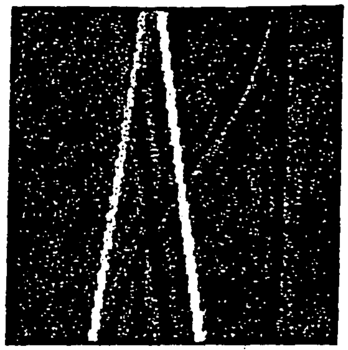


Figure 23a

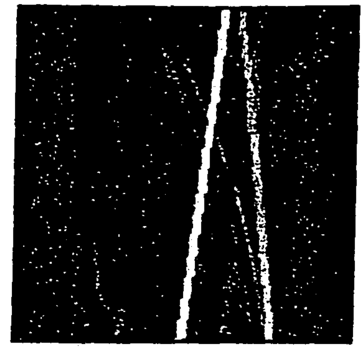


Figure 23b

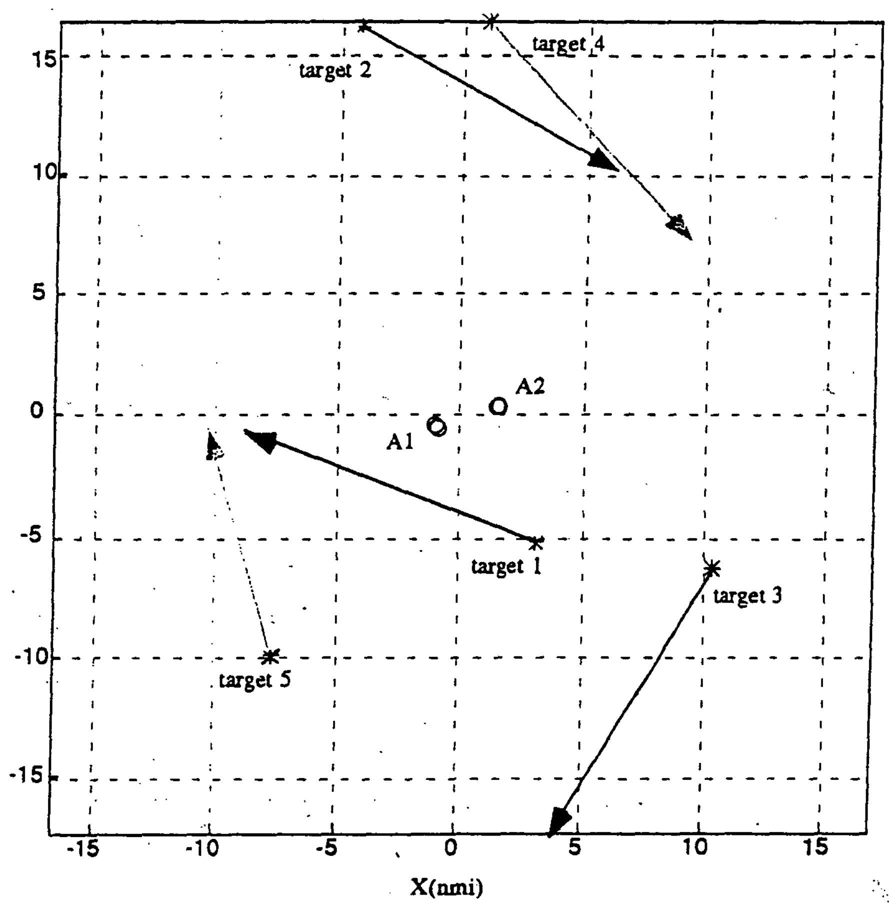


Figure 24

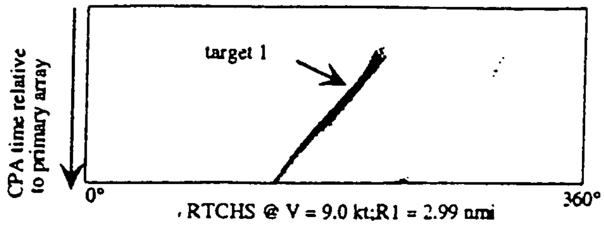


Figure 25a

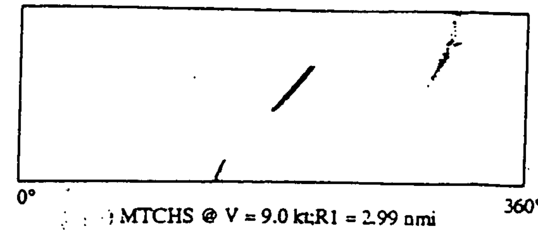


Figure 25b

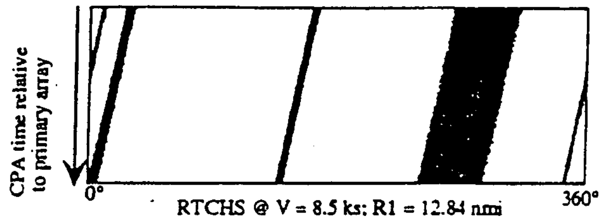


Figure 25c

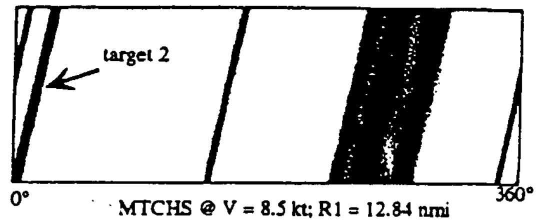


Figure 25d

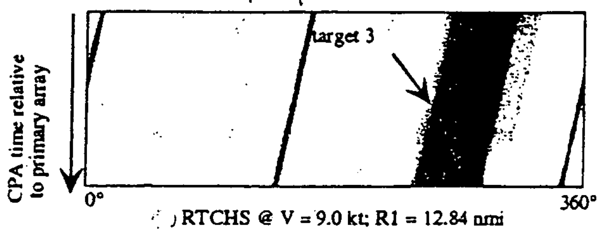


Figure 25e

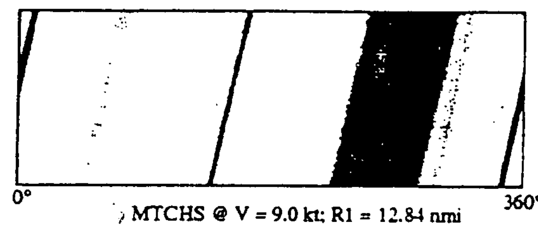


Figure 25f

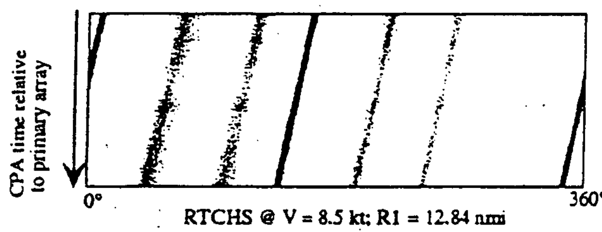


Figure 25g

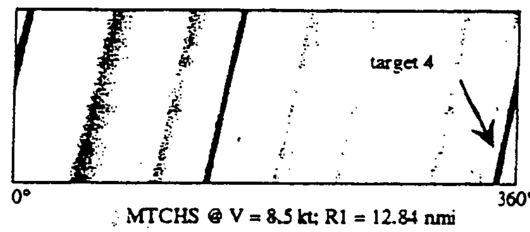


Figure 25h

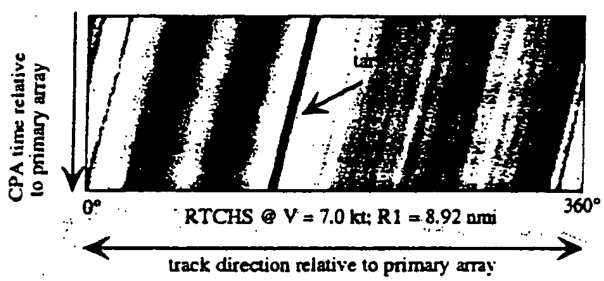


Figure 25i

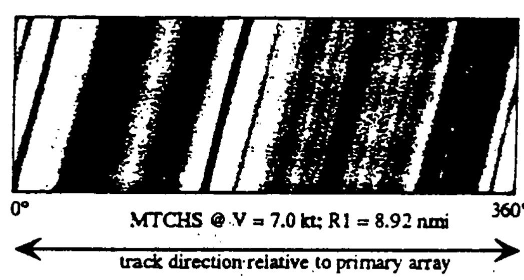


Figure 25j

A basic Flow Chart for the Additive Composite Hough Transform

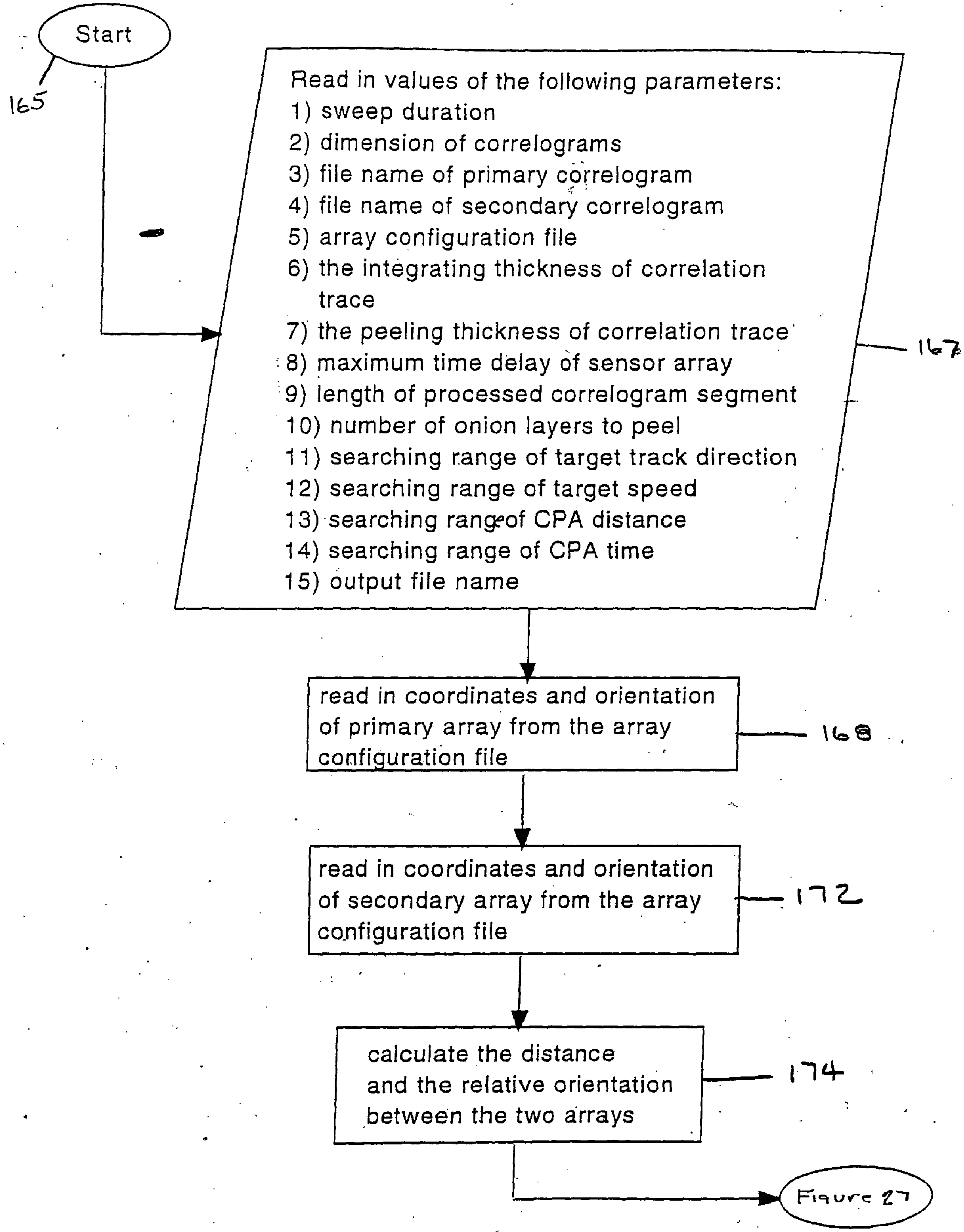


Figure 26

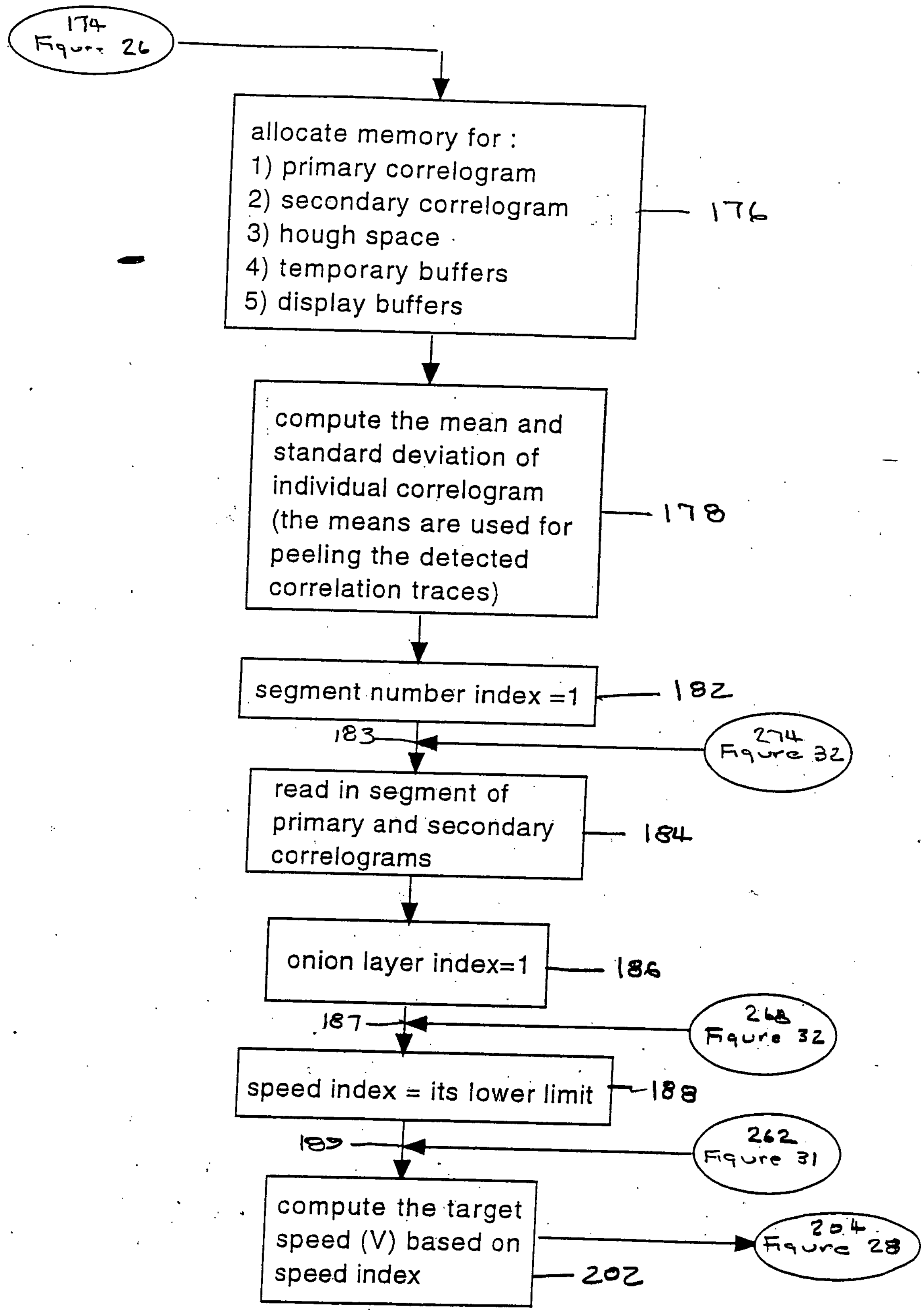

174
Figure 26
allocate memory for :
1) primary correlogram
2) secondary correlogram
3) hough space
4) temporary buffers
5) display buffers
176
compute the mean and standard deviation of individual correlogram (the means are used for peeling the detected correlation traces)
178
segment number index =1
182
183
274
Figure 32
read in segment of primary and secondary correlograms
184
onion layer index=1
186
187
268
Figure 32
speed index = its lower limit
188
189
262
Figure 31
compute the target speed (V) based on speed index
202
204
Figure 28

Figure 27

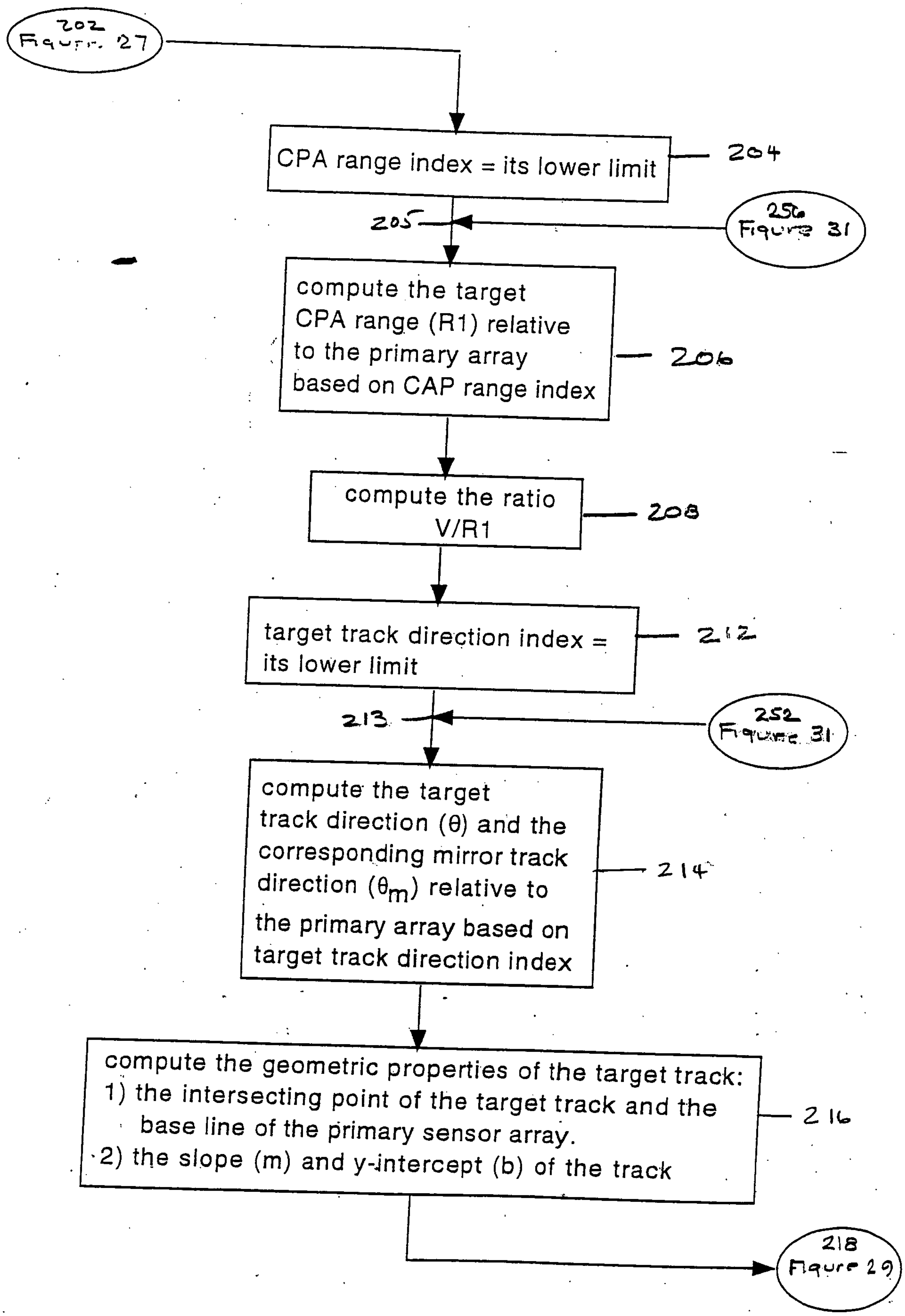

202 Figure 27
CPA range index = its lower limit
204
205
256 Figure 31
compute the target CPA range (R1) relative to the primary array based on CAP range index
206
compute the ratio V/R1
208
target track direction index = its lower limit
212
213
252 Figure 31
compute the target track direction (θ) and the corresponding mirror track direction (θm) relative to the primary array based on target track direction index
214
compute the geometric properties of the target track:
1) the intersecting point of the target track and the base line of the primary sensor array.
2) the slope (m) and y-intercept (b) of the track
216
218 Figure 29

Figure 28

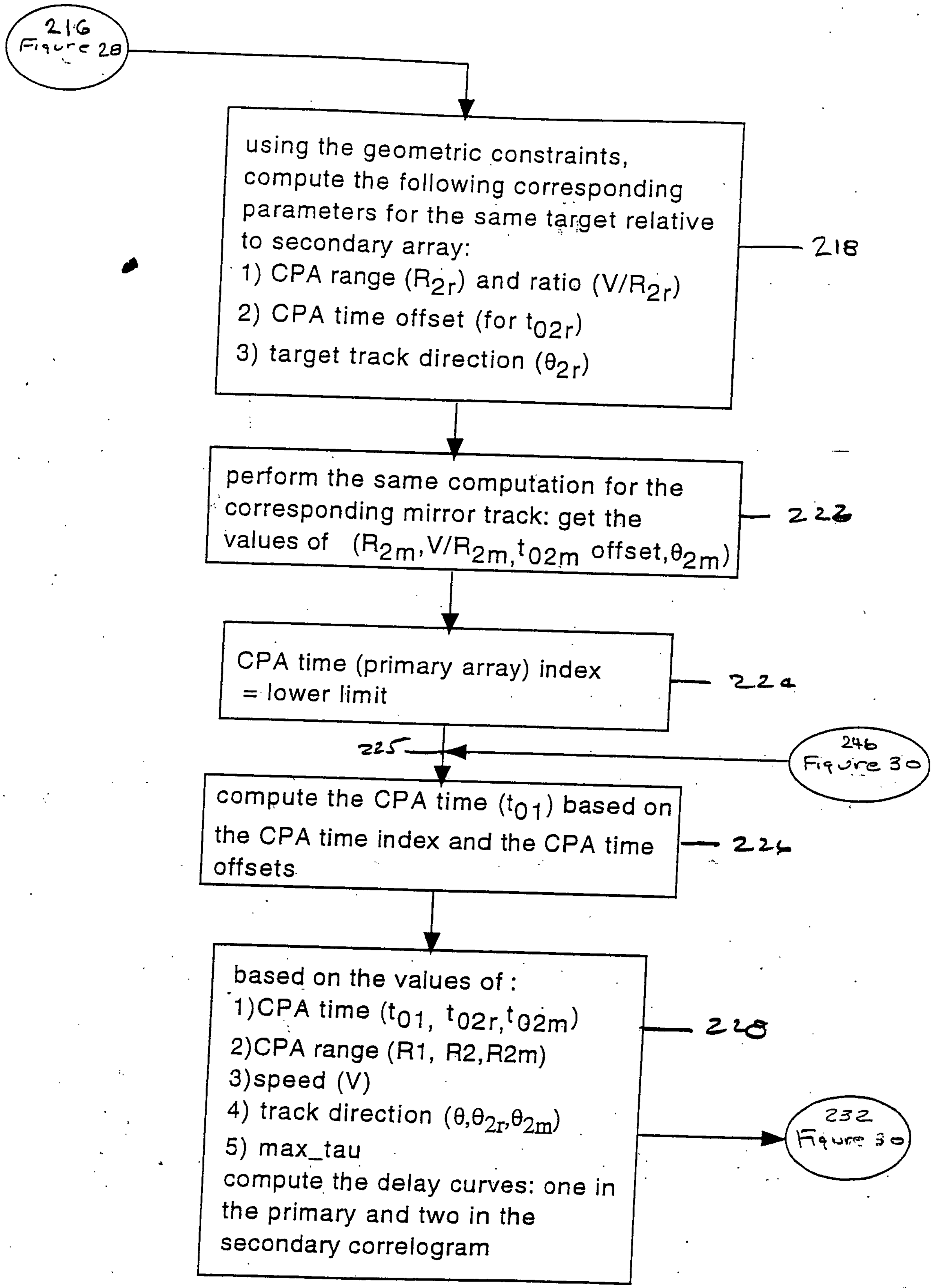

216 Figure 28
using the geometric constraints, compute the following corresponding parameters for the same target relative to secondary array:
1) CPA range (R_{2r}) and ratio (V/R_{2r})
2) CPA time offset (for t_{02r})
3) target track direction (θ_{2r})
218
perform the same computation for the corresponding mirror track: get the values of (R_{2m}, V/R_{2m}, t_{02m} offset, θ_{2m})
222
CPA time (primary array) index = lower limit
224
225
246 Figure 30
compute the CPA time (t_{01}) based on the CPA time index and the CPA time offsets
226
based on the values of :
1)CPA time (t_{01}, t_{02r}, t_{02m})
2)CPA range (R1, R2,R2m)
3)speed (V)
4) track direction (θ, θ_{2r}, θ_{2m})
5) max_tau
compute the delay curves: one in the primary and two in the secondary correlogram
228
232 Figure 30

Figure 29

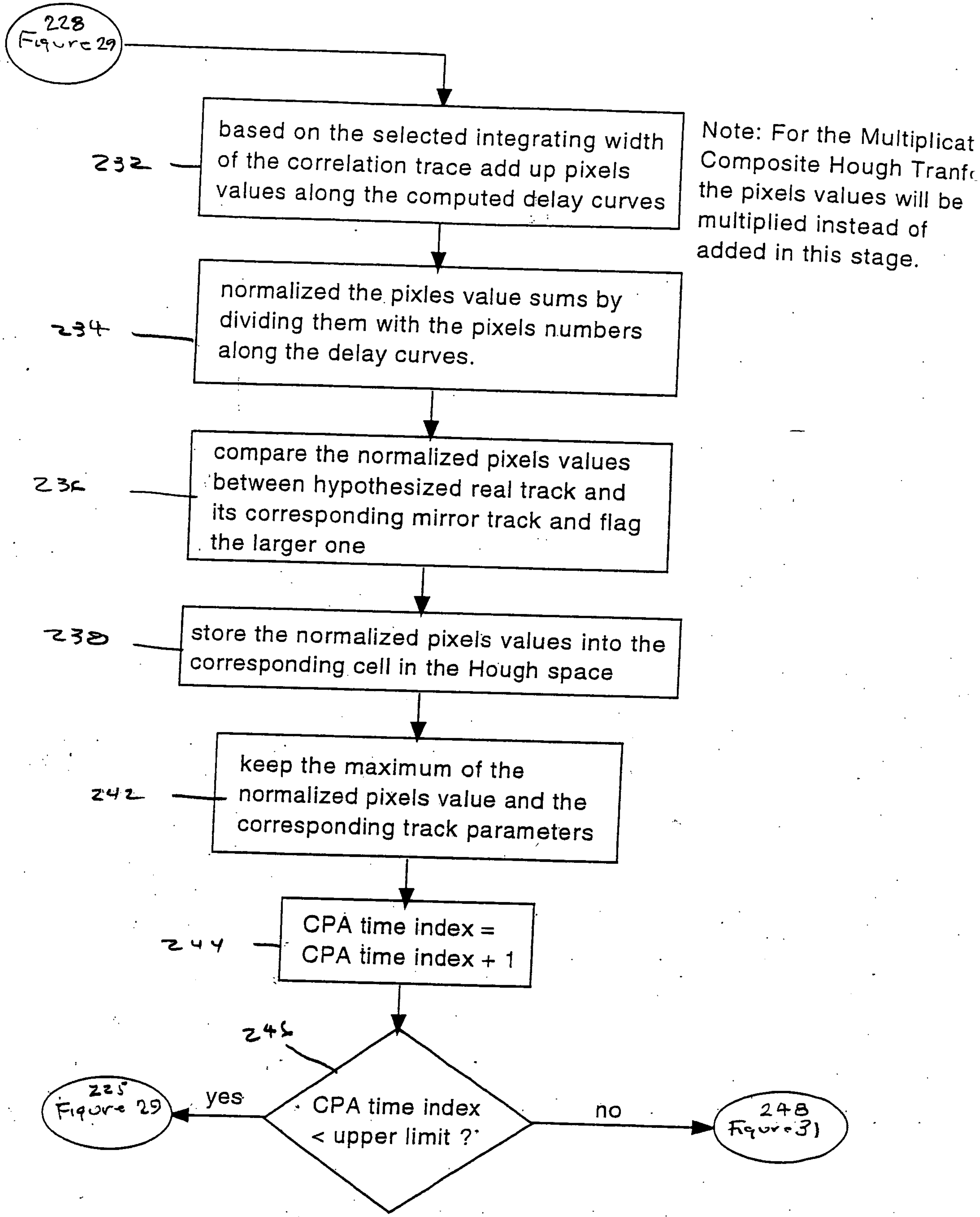

228
Figure 29
based on the selected integrating width of the correlation trace add up pixels values along the computed delay curves
232
Note: For the Multiplicat
Composite Hough Tranf
the pixels values will be multiplied instead of added in this stage.
normalized the pixles value sums by dividing them with the pixels numbers along the delay curves.
234
compare the normalized pixels values between hypothesized real track and its corresponding mirror track and flag the larger one
236
store the normalized pixels values into the corresponding cell in the Hough space
238
keep the maximum of the normalized pixels value and the corresponding track parameters
242
CPA time index =
CPA time index + 1
244
246
CPA time index
< upper limit ?
yes
225
Figure 29
no
248
Figure 31

Figure 30

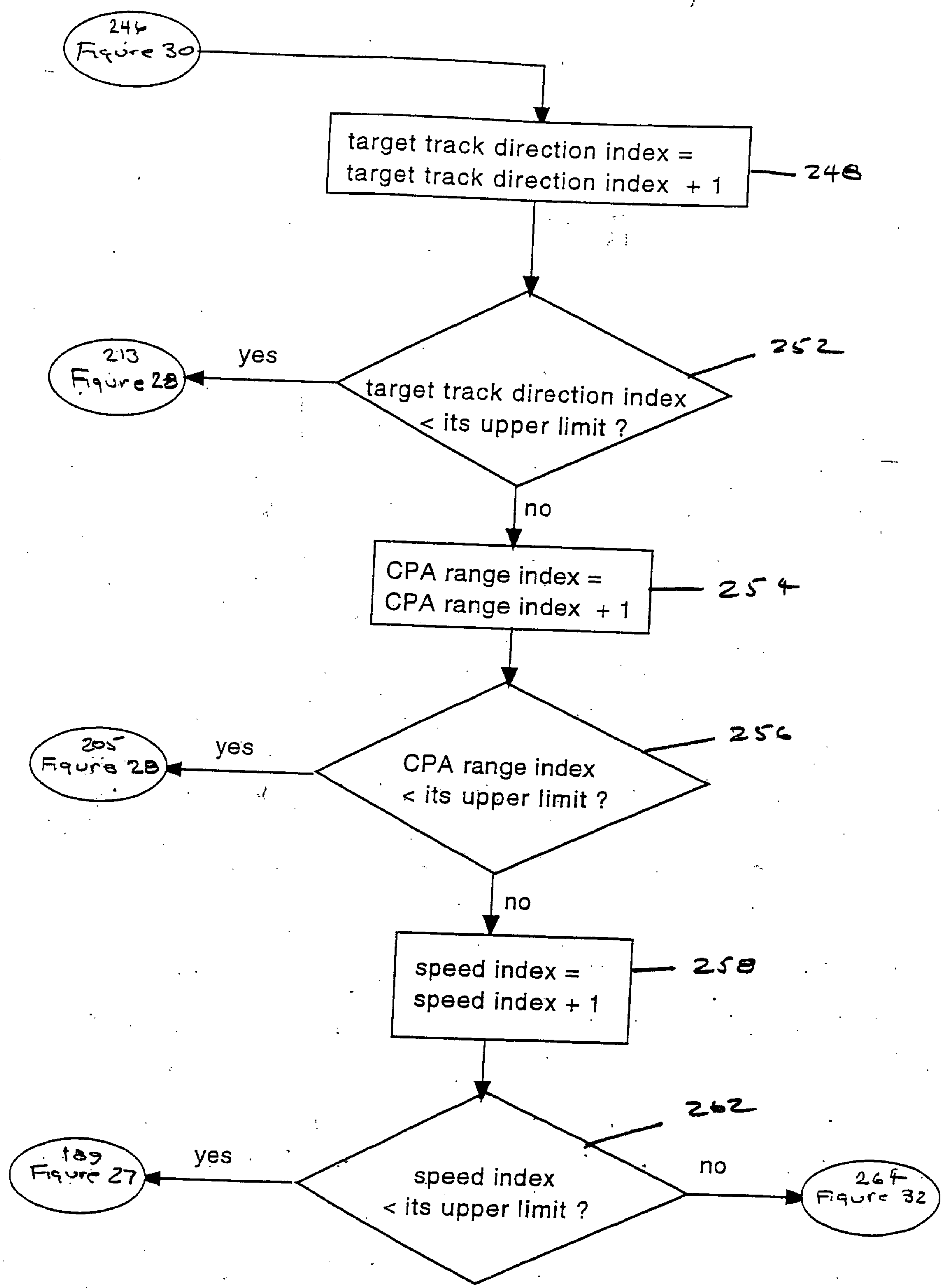

246
Figure 30
target track direction index =
target track direction index + 1
248
target track direction index
< its upper limit ?
252
yes
213
Figure 28
no
CPA range index =
CPA range index + 1
254
CPA range index
< its upper limit ?
256
yes
205
Figure 28
no
speed index =
speed index + 1
258
speed index
< its upper limit ?
262
yes
189
Figure 27
no
264
Figure 32

Figure 31

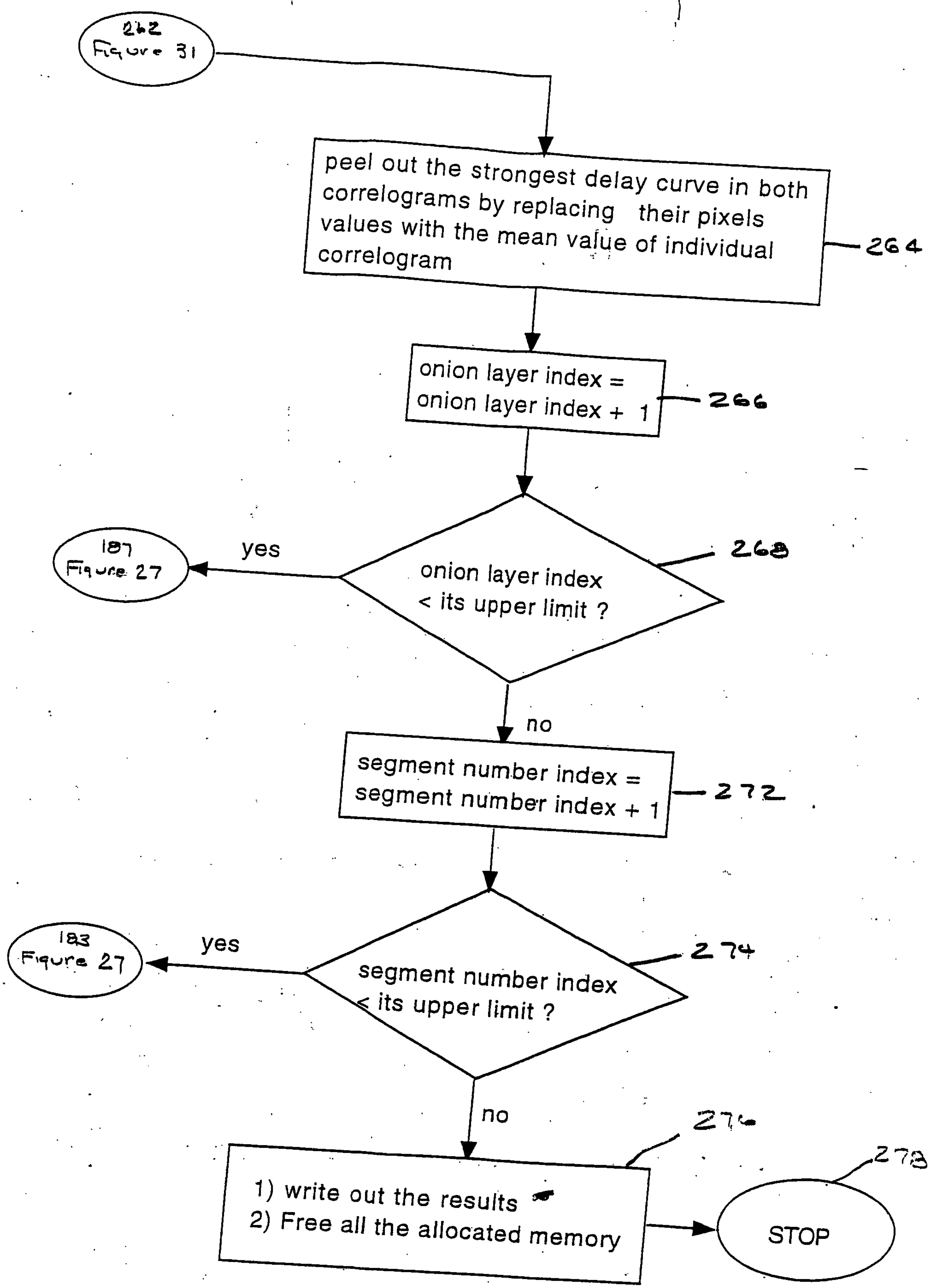

262
Figure 31
peel out the strongest delay curve in both correlograms by replacing their pixels values with the mean value of individual correlogram
264
onion layer index =
onion layer index + 1
266
187
Figure 27
yes
onion layer index
< its upper limit ?
268
no
segment number index =
segment number index + 1
272
183
Figure 27
yes
segment number index
< its upper limit ?
274
no
276
1) write out the results
2) Free all the allocated memory
278
STOP

Figure 32